Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Journal of Computational Physics 190 (2003) 229–248

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Differential equation-based wall distance computation for DES and RANS

P.G. Tucker *

Department of Engineering, Fluid Dynamics Research Centre, University of Warwick, Coventry CV4 7AL, UK

Received 21 October 2002; received in revised form 27 January 2003; accepted 12 May 2003

Abstract

Surprisingly expensive to compute wall distances are still used in a range of key turbulence and peripheral physics models. Potentially economical, accuracy improving differential equation-based distance algorithms are considered. These involve elliptic Poisson and a hyperbolic-natured Eikonal equation approaches. Numerical issues relating to non-orthogonal curvilinear grid solution of the latter are addressed. Eikonal extension to a Hamilton–Jacob equation is discussed. Use of this extension to improve turbulence model accuracy and, along with the Eikonal, enhance detached eddy simulation (DES) techniques is considered. Application of the distance approaches is studied for various geometries. These include a plane channel flow with a wire at the centre, a wing–flap system, and a supersonic double-delta configuration. Although less accurate than the Eikonal, Poisson method-based flow solutions are extremely close to those using a search procedure. For a moving grid case the Poisson method is found especially efficient. Results show that the Eikonal equation can be solved on highly stretched, non-orthogonal, and curvilinear grids. A key accuracy aspect is that metrics must be upwinded in the propagating front direction. The Hamilton–Jacobi equation is found to have qualitative turbulence model improving properties.

1. Introduction

Wall distances, d, are still a primary parameter in a range of key turbulence models [1–9] and also peripheral applications incorporating additional solution physics [10]. They are also an essential element in numerous two equation k–ε [11], zonal Reynolds Averaged Navier–Stokes [12], and zonal LES (large Eddy simulation) or DES [13] approaches. For turbulence models, d is just required close to walls. The SA (Spalart–Allmaras) model [4] is probably the most extreme requiring d accurate for a third of the boundary layer. Surprisingly, for highly optimised RANS/URANS (unsteady Reynolds averaged Navier–Stokes) solvers, the effort in calculating d can be a significant fraction of the total solution time. For example, even with Cray C90 class computers it can take 3 h just to gain d [14]. For flows with time-dependent geometry (such as Computational

[*] Tel.: +44-24-7652-8242; fax: +44-24-7641-8922.
E-mail address: p.tucker@warwick.ac.uk.

0021-9991/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0021-9991(03)00272-9

Aeroelasticity) or mesh refinement, clearly, this feature is exacerbated [15]. Because of d evaluation expense in some codes dangerous approximations are made. The following are given by Spalart [16]:
(a) computing distances down grid lines and not allowing for grid non-orthogonality;
(b) computing 'd' as the distance between a field point and the nearest wall grid point; and
(c) in multiblock solutions ascertaining 'd' on a purely block-wise basis ignoring the possibility that the nearest wall distance might be associated with another block.
The latter can create large inaccuracies and also non-smooth, unhelpful to convergence d distributions. In relation to point (c), Spalart [16] notes that for overset grids the situation can arise where the same point in space has different equations.

Clearly the listed practices will give mostly unhelpful inexact distances (or even multiple distances) $\tilde{d}$. However, the careful modification of d to some $\tilde{d}$ can remedy turbulence model deficiencies or extend modelling potential. For example, in the SA and v_{t92} [6] models roughness can be accounted for by a distance displacement d_0 such that $\tilde{d} = d + d_0$. Also, if $\tilde{d} = d$ sharp convex features such as a thin wire (referred to here as the '*thin wire problem*') or wing trailing edge can have disproportionate turbulence influences [17]. For a thin wire in a channel the anomalous situation arises where in the wall normal direction the wire (no matter how small) has just as strong a turbulence damping influence as the channel walls (see later). Wall proximity reduces eddy viscosity through boosting turbulence destruction terms. Hence, the excessive influence of sharp convex features can be lessened by ensuring $\tilde{d} > d$. For corners or bodies/surfaces in close proximity the increased multiple surface turbulence damping effect [8,17–19] should be taken into account. Setting $\tilde{d} < d$ is a convenient mechanism for achieving this.

Distance evaluation methods can be broadly classified under the following headings: (I) search procedures, (II) integral approaches, and (III) differential equation-based methods. Crude search procedures require $\mathrm{O}(n_v n_s)$ operations where n_s and n_v correspond to the number of surface and internal node points. Wigton [14] and Boger [15] present more efficient search procedures needing $\mathrm{O}(n_v\sqrt{n_s})$ and $\mathrm{O}(n_v \log n_s)$ operations, respectively. Integral approaches are described in [14,19]. For complex geometries they are difficult to apply. Hence, the focus here is on differential equation-based methods. Advantageously, these are suitable for vector and parallel computers. Differential equation methods are discussed below.

1.1. Differential equation-based distance methods

1.1.1. Level set method

Osher and Sethian [20] use the *level set* method. This solves the following equation in an *initial value framework*

$$\frac{\partial \phi}{\partial t} + |\nabla \phi| = \Gamma \nabla^2 \phi. \tag{1}$$

The d is evaluated from the time at which ϕ reaches a 'set level' say $\phi = 0$. The $\phi = 0$ level could be considered as a propagating front. The exact wall distance is gained as $\Gamma \to 0$. In [20], the Eq . (1) solution needs $\mathrm{O}(n_v^4)$ operations.

1.1.2. Poisson equation method

A Poisson equation (see [21,22]) d method proposed by Spalding can also be used. For this, in Eq. (1), $-\partial\phi/\partial t = \Gamma = 1$, $|\nabla\phi| = 0$, and gradients of ϕ are used to gain d. For curved surfaces the following L_2 norm-based approximation is recommended

$$d = \pm\sqrt{\sum_{j=1,3}\left(\frac{\partial \phi}{\partial x_j}\right)^2} + \sqrt{\sum_{j=1,3}\left(\frac{\partial \phi}{\partial x_j}\right)^2 + 2\phi}. \tag{2}$$

The derivation of (2) includes the assumption that surfaces are extensive in the non-wall normal directions. Close to walls, this is a good approximation. In essence the Poisson method involves a straightforward elliptic boundary value problem solution. On solid boundaries ($\tilde{\Gamma}$) enclosing the domain Ω, $\phi = 0$. An efficient Poisson equation solver typically scales as $\mathrm{O}(n_v \log n_v)$. Here, the work of Tucker [21,22] is extended by exploring the use of the Poisson method for external flows and also curvilinear and moving grids.

1.1.3. Eikonal equation method

Motivated by the level set method solution expense, Sethain [23] uses a hyperbolically behaved boundary value formulation to the front propagation embodied in (1). This is given below

$$|\nabla\phi| = 1 + \Gamma\nabla^2\phi. \tag{3}$$

The time derivative now vanishes and $\phi = t$ – the front first *arrival time*. If $\Gamma = 0$, (3) becomes the Eikonal equation. With care this can be solved in $\mathrm{O}(n_v \log n_v)$ operations. For $\Gamma = 0$, the right-hand side of unity characterises the front propagation velocity. For this unit velocity $d = t$. Sethain solves Eq. (3) with $\Gamma \to 0$, i.e., just enough dissipation (arising through the numerical discretization) to gain an entropy (physically sensible) solution. Here the Laplacian is deliberately used to control the front propagation velocity, U, to gain $\tilde{d}$. The U can now be considered equal to $(1 + \Gamma\nabla^2\phi)^{-1}$. Eikonal equation solution in non-orthogonal coordinates is uncertain [24]. Hence, this aspect is also considered in the present work.

1.1.4. Navier–Stokes form of 'Eikonal equation'

Squaring the Eikonal equation ($|\nabla\phi| = 1$) and then adding a diffusion term gives the following:

$$(\nabla\phi)^2 = 1 + \Gamma\nabla^2\phi. \tag{4}$$

Now, if we define a vector

$$\mathbf{U} = \nabla\phi, \tag{5}$$

Eq. (4) can be rewritten in a non-conservative 'Navier–Stokes' form

$$\mathbf{U} \cdot \nabla\phi = 1 + \Gamma\nabla^2\phi. \tag{6}$$

Since $(\nabla\phi)^2 = \nabla \cdot (\phi\nabla\phi) - \phi\nabla^2\phi$ and $\mathbf{U} = \nabla\phi$ the following conservative form of (4) is also possible

$$\nabla \cdot (\phi\mathbf{U}) = 1 + \tilde{\Gamma}\nabla^2\phi, \tag{7}$$

where $\tilde{\Gamma} = \Gamma(1 + \phi)$. As is usual when solving such equations [23] to gain monotone solutions (a key wall distance realizability constraint) upwinding can be used for the left-hand side convective terms. Central differencing is applied to the diffusion terms. Considering Eq. (5) and that at walls $\phi = 0$ shows the upwind direction is away from walls. This suggests the essence of Sethian's marching approach where the Eikonal equation is solved by propagating d out from solid surfaces. The propagation can be stopped at a specified 'distance'. This can be chosen to correspond to the range where near wall distances are required in say DES or zonal RANS predictions. This aspect is explored further later. Again, the Eq. (6) Laplacian essentially modifies the front propagation velocity.

1.1.5. Laplacian form and role

Near a fine convex feature (wire) for theoretical correctness accurate distances are needed so $\tilde{d} = d$. However, to prevent excessive far field influence $\tilde{d} \gg d$. Since adjacent to a convex feature $\Gamma\nabla^2 d \gg 0$, with

Laplacian inclusion the desired effect of exaggerating d (i.e., delaying first arrival times) is naturally gained. Motivated by dimensional homogeneity, the need that as $d \to 0$, $\tilde{d} = d$ but $\nabla^2\phi \to \infty$ suggests

$$\Gamma = \varepsilon d, \tag{8}$$

where ε is a constant. Clearly more 'aggressive' functions than (8) (e.g., $\Gamma = \varepsilon(-1 + e^d)$) are possible but these are not explored. The Eikonal equation with $f(\phi)\nabla^2\phi$ added can formally be classified as a Hamilton–Jacobi equation.

At corners $\nabla^2 d < 0$, hence $\tilde{d} \ll d$. Therefore, with the Laplacian, the damping effects of 'extra' walls, discussed earlier, is naturally accounted for. At a d maximum, say for example at the centre line of a channel, again $\nabla^2 d < 0$. This is sensible, since again it will naturally model the damping effect of the two adjacent walls. As discussed later, the Laplacian perhaps has the potential to lessen non-smoothness issues [25] associated with SA DES. Eikonal equation distances will generally be discontinuous in gradient (this partly occurs where fronts propagating from different walls collide). Therefore, the $\nabla^2 d$ term smoothing has the potential to enhance convergence.

2. Numerical method

2.1. Transformed Eikonal equation

Discussion mostly focuses on modelling the more numerically challenging hyperbolic-natured terms outlined in Section 1. Transformation of the pure Eikonal equation (using the chain rule of differential calculus) into, for initial compactness, a two-dimensional curvilinear (ξ, η) system yields the following:

$$Q_\xi\left(\frac{\partial\phi}{\partial\xi}\right)^2 + Q_\eta\left(\frac{\partial\phi}{\partial\eta}\right)^2 = 1^2 + Q_{\xi\eta}\left(\frac{\partial\phi}{\partial\xi}\frac{\partial\phi}{\partial\eta}\right), \tag{9}$$

where, using for non-Q subscripted terms the shorthand form $\xi_x = \partial\xi/\partial x$, etc.

$$Q_\xi = \xi_x^2 + \xi_y^2, \tag{10a}$$

$$Q_\eta = \eta_x^2 + \eta_y^2, \tag{10b}$$

$$Q_{\xi\eta} = -2(\xi_x\eta_x + \xi_y\eta_y). \tag{10c}$$

The metrics are then as follows:

$$\xi_x = \frac{1}{|J|}\frac{\partial y}{\partial\eta}, \quad \eta_x = \frac{1}{|J|}\left(-\frac{\partial y}{\partial\xi}\right), \quad \xi_y = \frac{1}{|J|}\left(-\frac{\partial x}{\partial\eta}\right), \quad \eta_y = \frac{1}{|J|}\frac{\partial x}{\partial\xi}. \tag{11}$$

The determinant of the Jacobian in the above is

$$|J| = \left(\frac{\partial y}{\partial\eta}\right)\left(\frac{\partial x}{\partial\xi}\right) - \left(\frac{\partial y}{\partial\xi}\right)\left(\frac{\partial x}{\partial\eta}\right). \tag{12}$$

Clearly, it is also easy to transform the Laplacian (see for example Eq. (1)) into the (ξ, η) system. Importantly, a significant proportion of CFD (computational fluid dynamics) solvers, such as the NASA CFL3D code [26], ignore cross-derivatives thus assuming grid orthogonality. Applying this assumption to Eq. (9) would imply $Q_{\xi\eta} = 0$.

2.2. Differencing schemes

For the Eikonal equation (inviscid solution) upwind schemes are necessary. Following Sethian the first-order scheme due to Godunov is used (expressed this time in a three-dimensional form)

$$|\nabla\phi| \approx \Big\{ \max\left(\varDelta_{i-1,j,k}\phi, -\varDelta_{i+1,j,k}\phi, 0\right)^2 + \max\left(\varDelta_{i,j-1,k}\phi, -\varDelta_{i,j+1,k}\phi, 0\right)^2 + \max\left(\varDelta_{i,j,k-1}\phi, -\varDelta_{i,j,k+1}\phi, 0\right)^2 \Big\}^{1/2} = \mathrm{RHS}_{i,j,k}. \tag{13}$$

Assuming a transformed (ξ, η, ς) system

$$\varDelta_{i-1,j,k}\phi = \frac{\phi_i - \phi_{i-1}}{\Delta\xi}, \quad \varDelta_{i+1,j,k}\phi = \frac{\phi_{i+1} - \phi_i}{\Delta\xi}. \tag{14}$$

The subscripts refer to grid point locations. Eq. (13) ignores non-orthogonal terms. These must be treated in a manner consistent with (13). Higher-order versions of (13) are straightforward to implement. However, as will be seen later, in a turbulence modelling wall distance context, the consequences of the Eq. (13) diffusive tendencies are mostly beneficial. Where the right-hand side of (13) ($\mathrm{RHS}_{i,j,k}$) involves diffusion terms, standard second-order central differencing is used. A key issue, as will be seen later, is the differencing scheme used for the metrics embodied in Eqs. (11) and (12). For highly stretched grids, these must be formed using backwards/upwind differences. The direction of these is governed by the propagating front direction.

2.3. Pure Eikonal equation solution

The essence of Sethian's [23] fast marching method is illustrated using Fig. 1. In this, a 'wire' is represented on a Cartesian grid using a single node point. Points surrounding this are considered to be either seed, trial or those to be later solved. In the program developed here these points are labelled through the marker variable ns with the following respective values 2, 1, and 0. To start with, a seed (or multiple seeds) point is defined. The Eikonal equation is applied to assign this point a d value. The seed is represented in Frame (I) by the closed symbol. The Eikonal equation is then used for all immediate seed neighbours (trial points). These $ns = 1$ points are shown as open symbols. As shown in Frame (II), the trial point with the minimum d is taken as the next seed. Now, for this point, $ns = 2$. Then, as shown in Frame (III), d for all immediate neighbours to the new seed (for which $ns \neq 2$) are calculated. Hence the next seed point location is found. Frames (IV) and (V) show subsequent development stages for a clockwise moving circular front. In summary the procedure is as follows:
(a) Define a seed point.
(b) Solve for d at immediate neighbours to the seed (trial points).
(c) Make the trial point with the minimum d the next seed and return to (a).
This procedure is continued until $ns = 2$ for all points in Ω or inside a smaller domain/extent (as noted earlier, many turbulence models only need d to a maximum of one-third the boundary layer thickness) $\tilde{\Omega}$ identified by the Fig. 1 dotted line. In a practical system, all surface adjacent points can be taken as seeds and what is called an 'active front' produced.

2.4. Laplacian modelling and solution

The foregoing discussion has strongly focused on solution of the Eikonal equation. When for turbulence modelling purposes $\tilde{d}$ is needed (i.e., $\Gamma \neq 0$), the following solution procedure is used:
(a) Using frontal marching the Eikonal equation entropy solution is gained and then
(b) Eq. (3) with $\Gamma \neq 0$ is solved iteratively over Ω or $\tilde{\Omega}$.

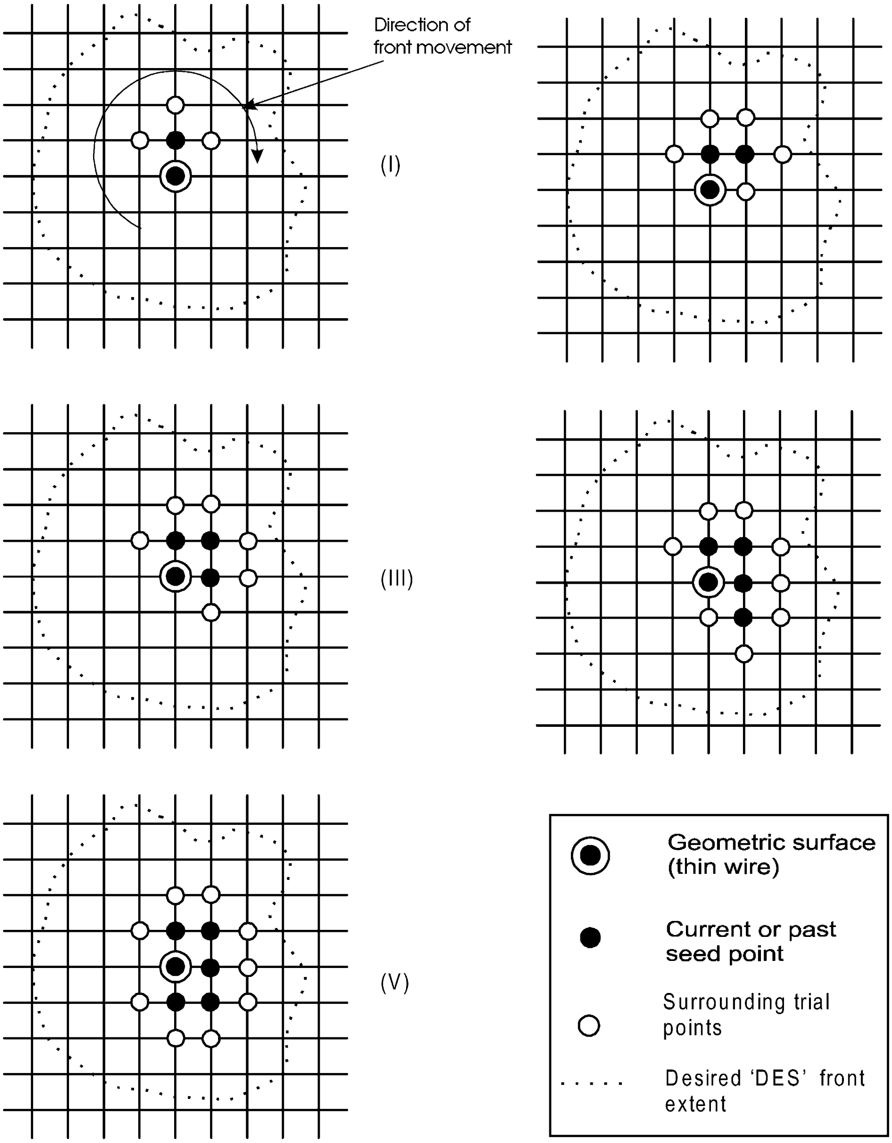
Direction of front movement
(I)
(II)
(III)
(IV)
(V)
◉ Geometric surface (thin wire)
● Current or past seed point
○ Surrounding trial points
..... Desired 'DES' front extent

Fig. 1. Eikonal equation solution using marching front approach with a single seed point.

For the larger part of Ω, $\nabla^2 d = 0$ and so the stage (a) solution is mostly a good initial guess. Therefore, a Newton solver (having quadratic convergence but a low convergence radius) is used for (b). However, at collision fronts, d gradients are discontinuous and the Laplacian locally large. Hence, Laplacian clipping is required where

$$-1 \leqslant \Gamma \nabla^2 d < C. \tag{15}$$

The lower limit is theoretical. Unlike with the level set method, the Eikonal equation does not permit backward front movements. The upper limit, C, is for stability. If the $\Gamma = f(d)$ function is strongly nonlinear a more robust solver than the Newton is likely to be required.

2.5. Zonal LES/DES Eikonal front extent

In the DES approach of Shur et al. [13] the SA distance scale is limited to

$$d < C_{\mathrm{DES}}\Delta_i, \tag{16}$$

where C_{DES} is a constant and, resorting for simplicity to a Cartesian co-ordinate system notation, the filter width for a node point/cell 'i' is

$$\Delta_i = \max\{\Delta x_i, \Delta y_i, \Delta z_i\}. \tag{17}$$

With the *k–l*-based zonal LES approach of Tucker and Davidson [27], the RANS/LES interface is at a specified y^+ location. Using an averaged (this would be spatial and if appropriate also temporal) global wall shear stress estimate the interface y^+ location can be defined in terms of a normal wall distance say $d_{\max}$. The global shear stress could be taken from an initial RANS or URANS prediction and subsequently refined. For

$$d < d_{\max}, \tag{18}$$

the *k–l* ($l \propto d$) RANS model is used and hence d is required. Outside the RANS region a filter scale must be specified. As noted earlier and in [25], DES can give length scale discontinuities. Usefully, the Hamilton–Jacobi method's Laplacian can perhaps help smooth these (see [28]).

2.6. Starting conditions

With the standard Eikonal-based DES approach, the following starting condition (initial guess) is sufficient:

$$\phi = C_{\mathrm{DES}}\Delta_i \quad \text{in } \Omega. \tag{19}$$

The d computation will naturally terminate when, at the active front, $d \geqslant C_{\mathrm{DES}}\Delta_i$. For most DES implementations d is needlessly evaluated for all Ω. With the current approach this wasted computational effort is avoided. Also, the d array is such that, if read into a sufficiently accurate URANS solver, with the SA model activated, a DES solution will naturally arise. No code modifications or needless intrinsic arguments are needed. For Eikonal-based RANS, zonal RANS, and LES [27] the starting condition below can be used

$$\phi = d_{\max} \quad \text{in } \Omega. \tag{20}$$

With *zonal* RANS/LES $d_{\max}$ could correspond to a prescribed model interface. For RANS economy, $d_{\max}$ could characterise the near wall turbulence destruction term activity range. Here, for *internal* flows without zonal modelling $d_{\max} \to \infty$ is used. This is a safe default value. For Poisson method predictions $\phi = 0$ in Ω is adequate.

2.7. Boundary conditions

At smooth solid walls, the following Dirichlet condition is applied:

$$\phi = 0 \quad \text{on } \tilde{\Gamma}. \tag{21}$$

At flow/far field boundaries

$$\frac{\partial \phi}{\partial n} = 0 \quad \text{on } \tilde{\Gamma} \tag{22}$$

can be used, where n is the boundary normal co-ordinate. However, if Ω is sufficiently large, (21) makes a stable far field condition. This is used here for the Poisson method.

Eikonal-based equations are tested for overset and abutted grids. For the abutted, on non-solid surface block boundaries differential conditions (Eq. (22)) are, due to the interface nature, found adequate. For the overset, a mono-block grid Eikonal solution is represented on a dual block flow solution grid (more details are given later). The above two approaches avoid generating two quite different multi-block Eikonal solvers.

2.8. General solution information

When evaluating Eq. (2), away from solid walls second-order central differences are used. For simplicity, with multiblock solutions at block boundaries first-order backward differences are implemented. The Poisson method simultaneous equations are solved using a crude ADI procedure with under-relaxation incorporated through a pseudo time term. For time invariant grids multigrid interpolation operators are used to automatically overlay solutions from coarser grids. Eikonal equation numerical solution requires a series of squaring and square root operations [29]. Therefore, attention to arithmetic precision is important. Modified versions of two structured grid, CFD solvers are used. These are the NASA CFL3D [26] and NTS [13] codes. Curvilinear grid Eikonal solutions generally use iteration.

3. Discussion of results

The following Fig. 2 geometries are considered:

(a) square box with thin wire;
(b) plane channel with and without a thin wire;
(c) wing with flap;
(d) double delta wing configuration.

3.1. Square box with thin wire (Case (a))

Figs. 3–5 give Eikonal-related Case (a) parameters. Fig. 3, frame (a) gives Eikonal distances. Frame (b) shows the Laplacian behaviour for an Eq. (8)-based Hamilton–Jacobi $\varepsilon = 0.2$ solution. The Laplacian is positive near the wire and negative (hence dashed contour lines) at corners. Fig. 4 shows the Laplacian influence on d near the wire. Frames (a) and (b) give d for Eikonal and Hamilton–Jacobi solutions, respectively. Comparing these frames shows that the Laplacian exaggerates d thus reducing the 'thin wire problem'. Since the wire has been approximated as a single node, on a Cartesian grid, the erroneous Godunov scheme trait of flattening the propagating front, noted by Tsai [29], is especially clear. Fig. 5 shows the Laplacian influence in corners. Frames (a) and (b) are for Eikonal and Hamilton–Jacobi solutions, respectively. Frame (b) shows the Laplacian correctly (see DNS data [18]) allows the turbulence model to sense the damping influence of both corner walls. This can be necessary to predict Prandtl's motions of the second kind (see [10]).

3.2. Plane channel flow (Case (b))

Fig. 6 gives Eikonal and Hamilton–Jacobi-based Case (b) distances when the channel has no wire. The full and dashed lines are Eq. (3) solutions with $\Gamma \neq 0$ (Hamilton–Jacobi solution) and $\Gamma = 0$, respectively. To make the Frame (a) Laplacian smoothing influence clear $\varepsilon = 2$, i.e., a large value. The resulting reduced

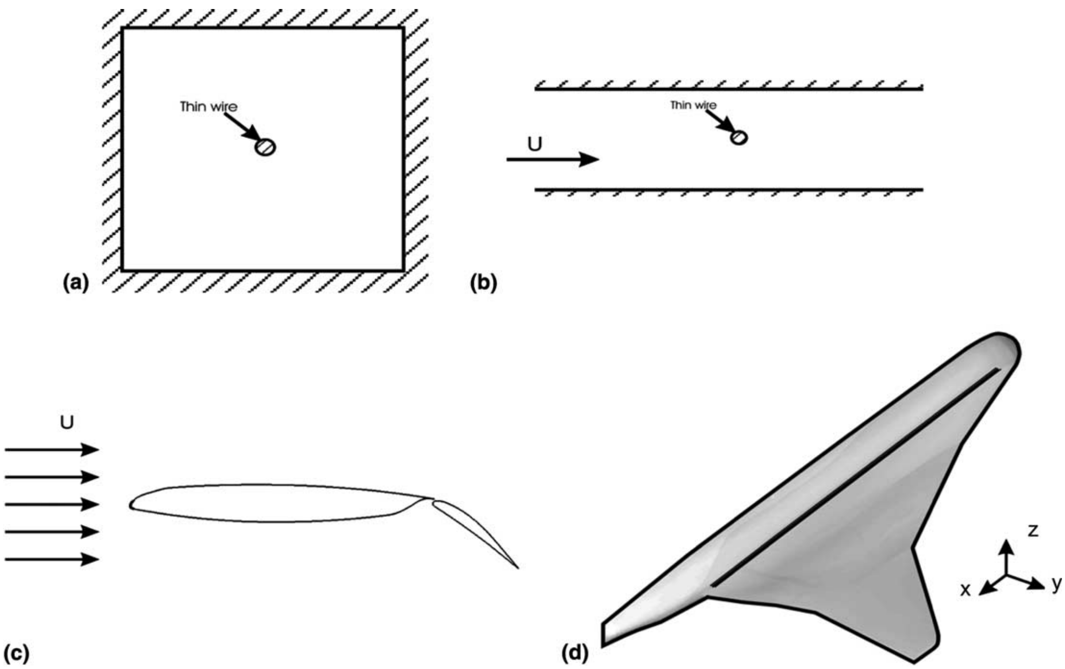

Fig. 2. Geometries considered: (a) square box with thin wire; (b) plane channel with optional thin wire; (c) wing and flap and (d) double-delta geometry.

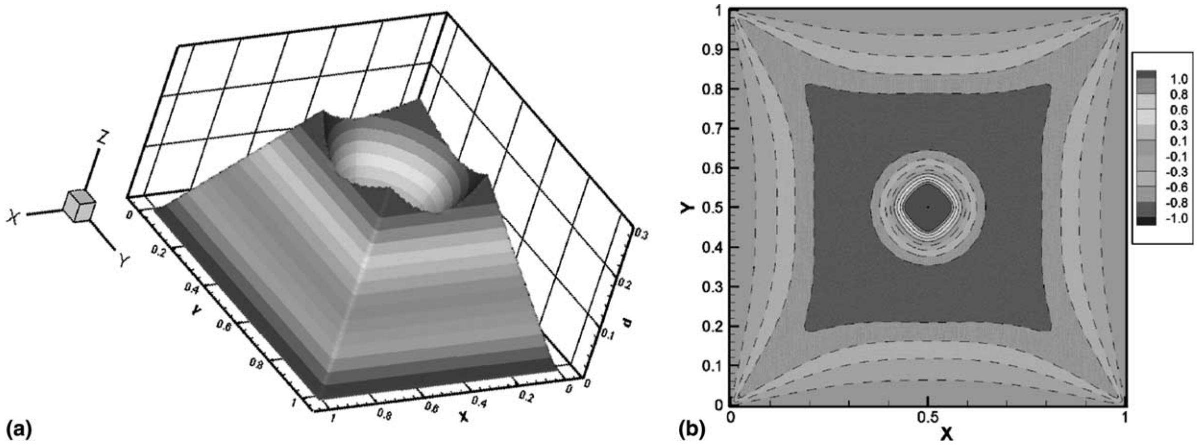

Fig. 3. Distance-related parameters for a thin wire in a square domain: (a) Eikonal equation d solution and (b) Hamilton–Jacobi Laplacian distribution.

d scale indicates the turbulence damping influence of both walls is now sensed [17]. For Case (b) the Poisson method will give exact distances. It is designed to do this.

Flow solutions for $Re = 1 \times 10^5$ (based on the full channel width) and a fully developed inlet profile are now considered. A wire having a diameter of 1/40th the full channel width is placed at the channel centre. Fig. 7 gives the NTS over-set flow solution mesh and also SA model turbulent viscosity contours. The

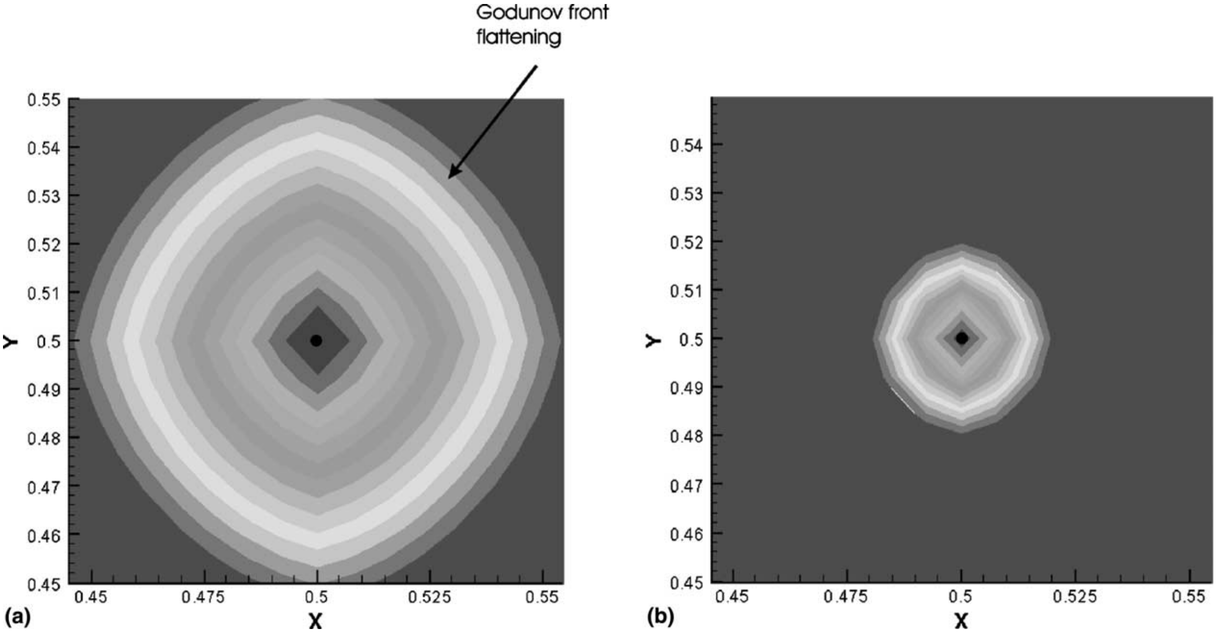

Fig. 4. Influence of Laplacian on d around a thin wire: (a) Eikonal solution (no Laplacian) and (b) Hamilton–Jacobi solution (Laplacian included).

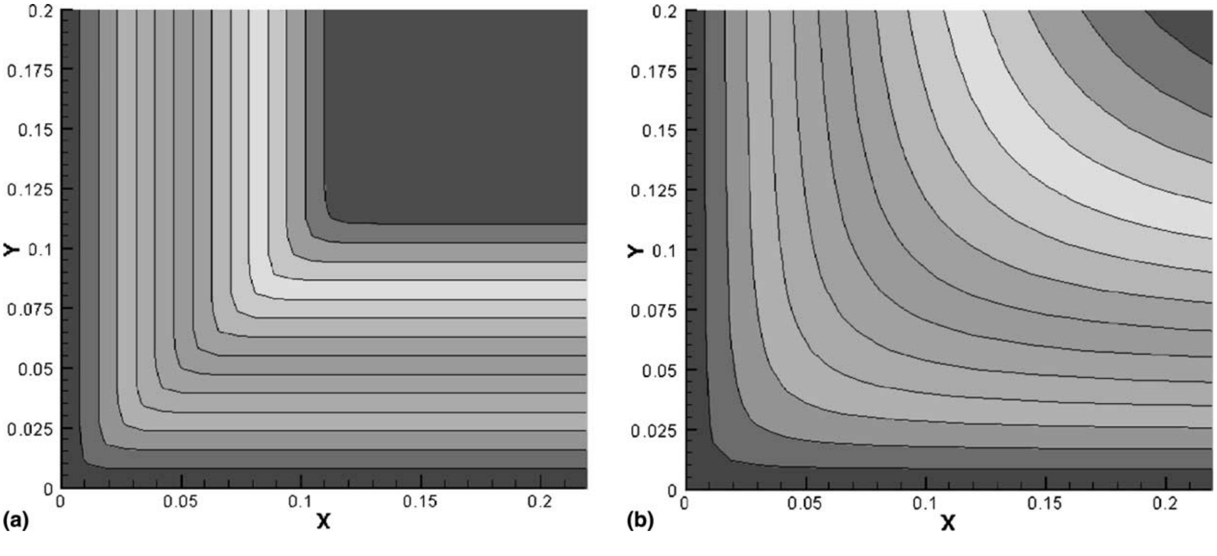

Fig. 5. Influence of Laplacian on d at corners: (a) Eikonal solution (no Laplacian) and (b) Hamilton–Jacobi solution (Laplacian included).

numerical/modelling accuracies and case set-up (symmetry at the $Y = 0$ plane) ensure there is no vortex shedding. Hamilton–Jacobi computed distances (evaluated using the approach of Tucker and Pan [30] and not over-set grids) are used to modify d just in the cylindrical Fig. 7(a) grid block. Figs. 7(a) and (b) give

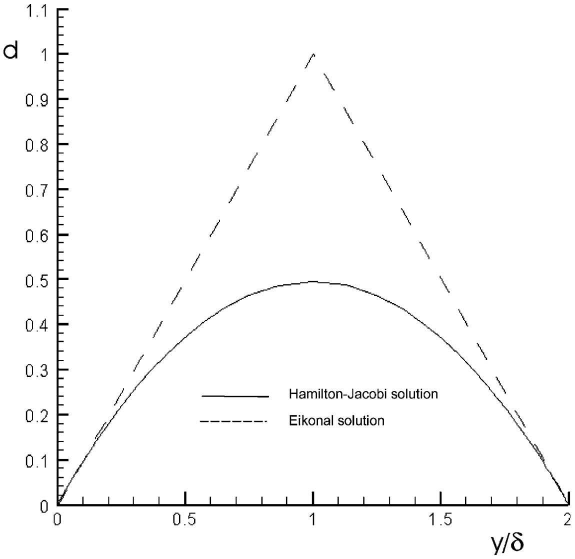

Fig. 6. Plane channel Eikonal and Hamilton–Jacobi distances.

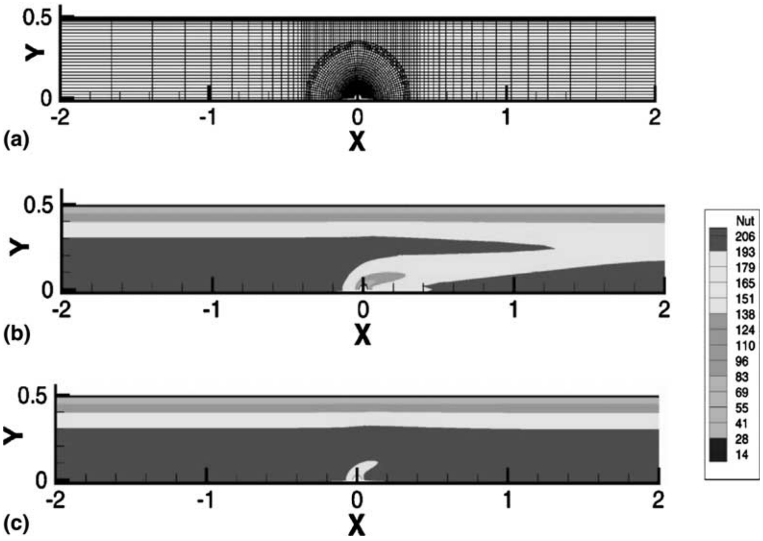

Fig. 7. Overset solution mesh and turbulent viscosity contours: (a) mesh; (b) Eikonal turbulent viscosity; and (c) Hamilton–Jacobi turbulent viscosity.

turbulent viscosity contours for Eikonal and Hamilton–Jacobi solutions, respectively. For illustrative purposes these turbulent viscosity plots use $\varepsilon \approx 5$. Clearly, the Hamilton–Jacobi approach has reduced the excessive wire influence. A factor of 10 reduction in the wire diameter gives just as strong a wire influence on solutions. However, caution is required. If ε is too large, although the wire disruption in the channel core is reduced, the wall shear stress is excessively contaminated. For overall flow improvements $\varepsilon < 1$ is needed. Further studies into possible Γ functions will hopefully be carried out as part of future work.

3.3. Wing with flap (Case (c))

In a zonal LES [27] context (for $y^+ < 250$), Fig. 8 shows Case (c) Eikonal d coloured grid features. As typical for high-speed flows, the grid is highly stretched in the wall normal direction. Although reasonably parallel to solid surfaces, grid lines are not always perpendicular.

Fig. 9 infers the **U** field (using search procedure distances) that should be gained when solving the Eikonal equation in a Navier–Stokes form Eqs. (6) and (7). Clarifying the front propagation nature of the Eikonal equation, Frame (a) shows velocities (**U**) are surface normal. Frames (b) and (c) show that in the light grey areas away from collision fronts $|\mathbf{U}| = 1$. At the darker zones gradients of d are

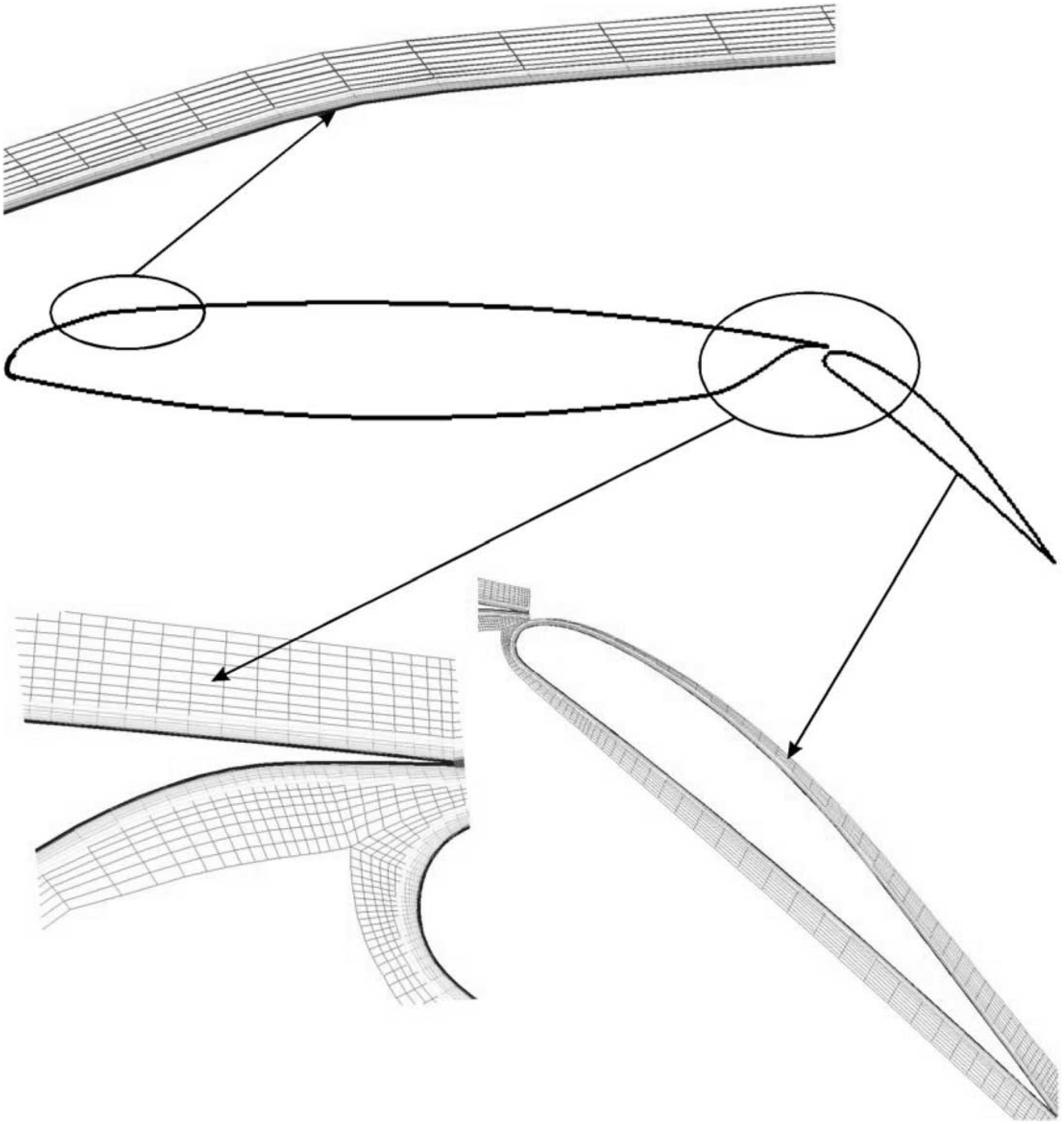

Fig. 8. Eikonal equation wall distance coloured grid features for wing–flap configuration.

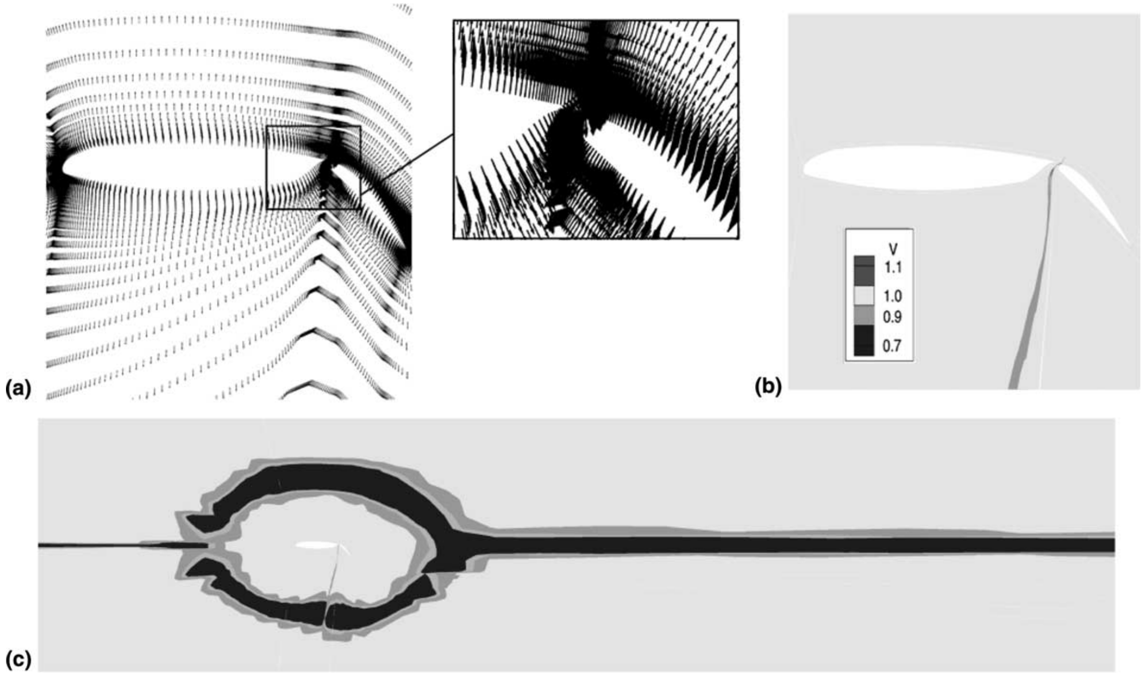

Fig. 9. Approximate Eikonal solution velocity field: (a) velocity vectors; (b) velocity magnitude focusing on wing and flap; and (c) velocity magnitude over complete domain.

discontinuous. These zones are analogous to compressible flow shocks. They also indicate boundaries for domains of influence.

A less numerically challenging equation than the Eikonal is the Poisson in the Poisson method. Fig. 10 shows the Case (c) Poisson equation solution (ϕ) field. The streamwise domain extent has been truncated. Upper and lower wind tunnel walls, accounted for in the flow simulations, are evident. Using Eq. (2) the Fig. 10 ϕ field can be converted into d. Fig. 11 qualitatively compares d contours for the search and Poisson

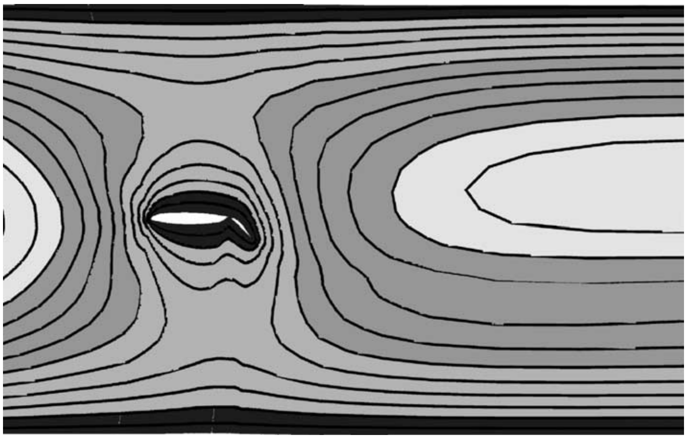

Fig. 10. Poisson equation solution contours for wing–flap configuration in a wind tunnel.

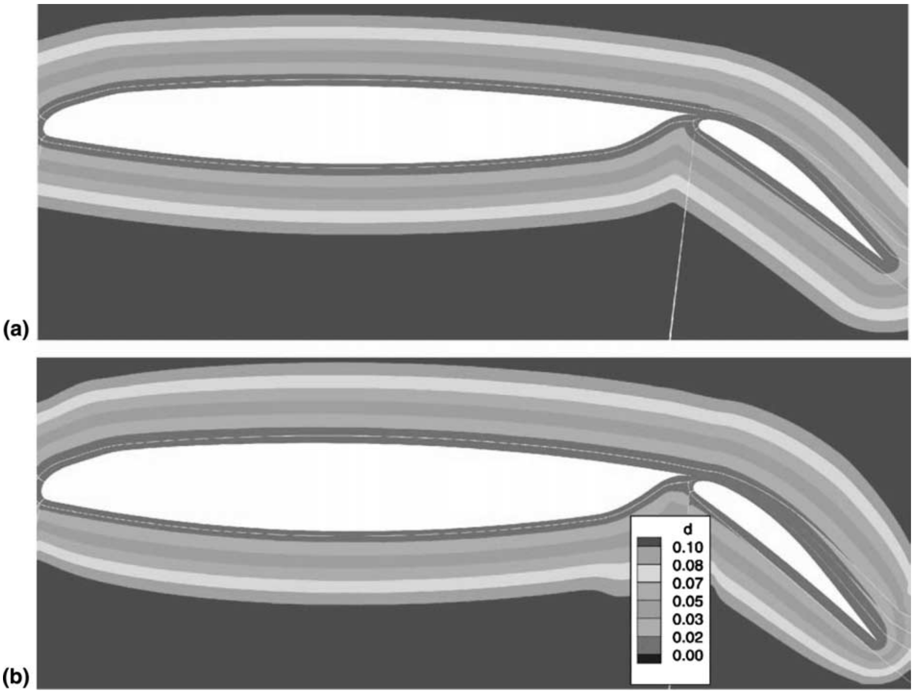

Fig. 11. Comparison of d for search procedure and Poisson method: (a) search procedure and (b) Poisson method.

methods. For the latter (frame (b)), there is unsightly d behaviour. This is due to significant levels of block interface grid non-orthogonality and the use of backward differences in (2) at these. Since the unsightly contours are away from walls this is purely a cosmetic issue.

Preliminary tests have been made directly solving (7). However, central differences where used in the convective terms. This resulted in divergence. For future work, it would seem important to use upwind differencing. As will be seen next, it is also important to use consistent differences in the metric evaluation. This would add significant additional coding complexity.

Fig. 12 shows top wing surface, mid-chord, Eikonal d results plotted against normal wall distance. Consistent with the geometry, the Eikonal solution is two-dimensional. The frame (a) and (b) vertical axes are d and % error, respectively. As can be seen, for highly stretched grids, using upwind-based metric differences that are consistent with the main discretization improves accuracy. Frame (a) shows, without metric upwinding, the d error grows (i.e., is additive) with surface distance.

The average, $y^+ < 400$, Eikonal equation error when solved with and without non-orthogonal grid terms is 0.8% and 2.7%, respectively. These values are for metric upwinding. Clearly, inclusion of the transformed equation cross-derivatives helps accuracy. Hence, the present work seems to clear uncertainty on whether the Eikonal equation can be solved in a non-orthogonal co-ordinate system. The average Poisson method error is 0.97%. However, the solution is made by altering the NASA CFL3D solver. This ignores non-orthogonal grid terms. As with the Eikonal equation, cross-term inclusion should improve Poisson method accuracy. Note, quoted errors are relative to d from the search procedure.

Fig. 13 gives $y^+ < 400$ error histograms for the Eikonal (with cross-derivatives) and Poisson-based approaches. Negative errors correspond to d over predictions. For both methods, conveniently, errors increase away from walls, where, for turbulence models, they matter less. Clearly, the Frame (a) Eikonal equation error distribution is most symmetrical. For the Poisson method (see Frame (b)), there is a d

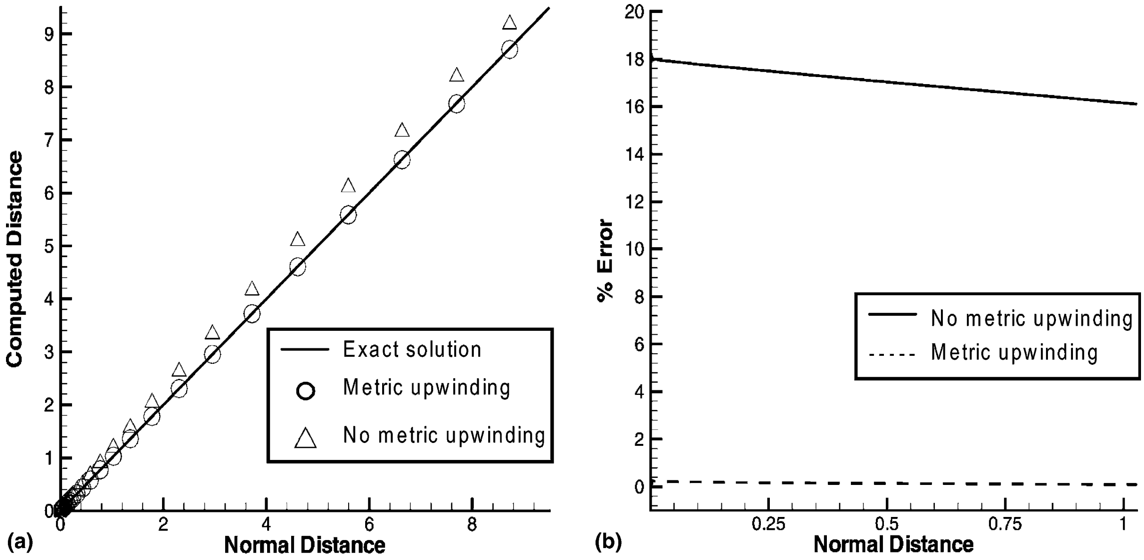

Fig. 12. Eikonal equation errors for solutions with and without Jacobian upwinding for wing–flap configuration: (a) computed *d* and (b) error in *d*.

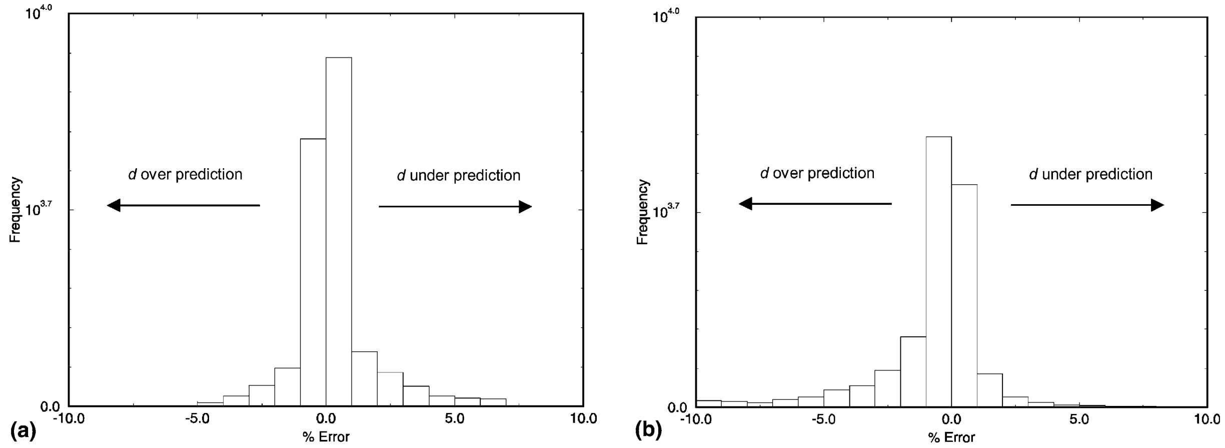

Fig. 13. Distance error histogram for wing–flap configuration: (a) Eikonal method and (b) Poisson method.

over-prediction trait. This is shown in Fig. 14 which plots the actual against modelled *d* downstream from the sharply convex flap trailing edge. The convex surface overestimation trait shown could be beneficial, minimizing the 'thin wire'-related problem. Another reason for the Poisson method's *d* over-predictions (distinct from the convex surface overestimation trait) is that grid orthogonality has been assumed. Hence, for highly skewed surface grids, although the notional wall 'normal' inter-nodal distance might be large the actual wall distance increment will be lower.

Fig. 15 gives Eikonal equation residuals for Eikonal and search procedure distances. Here, a positive residual corresponds to a *d* overestimate. Frame (a) is the residual for the actual Eikonal equation solution. The significant Frames (b) and (c) residuals are for the search procedure. Interestingly, (cf. Frames (a) and

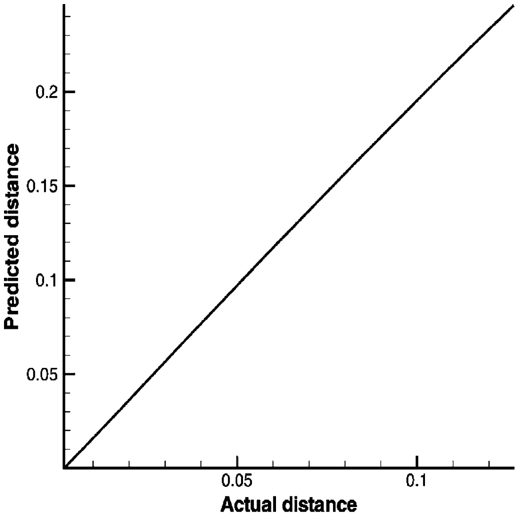

Fig. 14. Poisson method's helpful tendency to over-predict d around fine convex surfaces.

(b) regions (I) and (II)) there is some correspondence between the Eikonal and search procedure residuals. 'Error zone' (II) corresponds to a fairly abrupt change in wing surface contour. Due to the present scale this contour change is not easy to see. However, its effects are very evident in surface pressures. Zone (II) is where a forward wing element (when used) would meet the main wing. Possible reasons for the significant zone (II) errors and perhaps those in other regions is that:

(a) The grid has high aspect ratio streamwise orientated cells and so (without the use of splines) the surface normal direction and hence d is difficult to define.

(b) The Godunov approach (see [29]) flattens propagating fronts and hence the effects of abrupt changes in surface geometry will tend to be underestimated.

(c) Direct solution of the Eikonal equation involves repeated square and square root operations which with small wall normal grid spacings can give rise, through machine round of, to tangible errors (see [29]).

(d) As with any discrete numerical solution to a differential equation there will be grid convergence errors and due to the Eikonal equation nature other subtle numerical influences might be present (the issue of metric upwinding falls into this category).

(e) The Eikonal solution propagates fronts outward from solid surfaces, hence errors (which due to (b), (c), and (d) above, can be significant) sum as the front propagates.

As noted earlier, for this case, the Poisson method is less accurate than the Eikonal. However, CFL3D, SA model (which of all RANS models should show greatest d sensitivity) Poisson method lift and drag coefficients are within 0.05% of those for the search procedure. These values being for a solution with $Re = 23 \times 10^6$, based on the wing chord, and $Ma = 0.18$. Around 1 million cells give sensibly grid independent solutions.

3.4. Double delta wing (Case (d))

Fig. 16 gives the Case (d) Poisson method d error histogram. For this more complex configuration the average error is higher at 2.67%. However, as noted earlier the exact d is not always best. The Eikonal equation is not considered for this case.

A key Case (d) motivation is aeroelasticity studies. For details of the CFL3D mesh deformation approach, that freshly initialises the d search procedure for each moving mesh time-step, see [31]. Fig. 17 shows

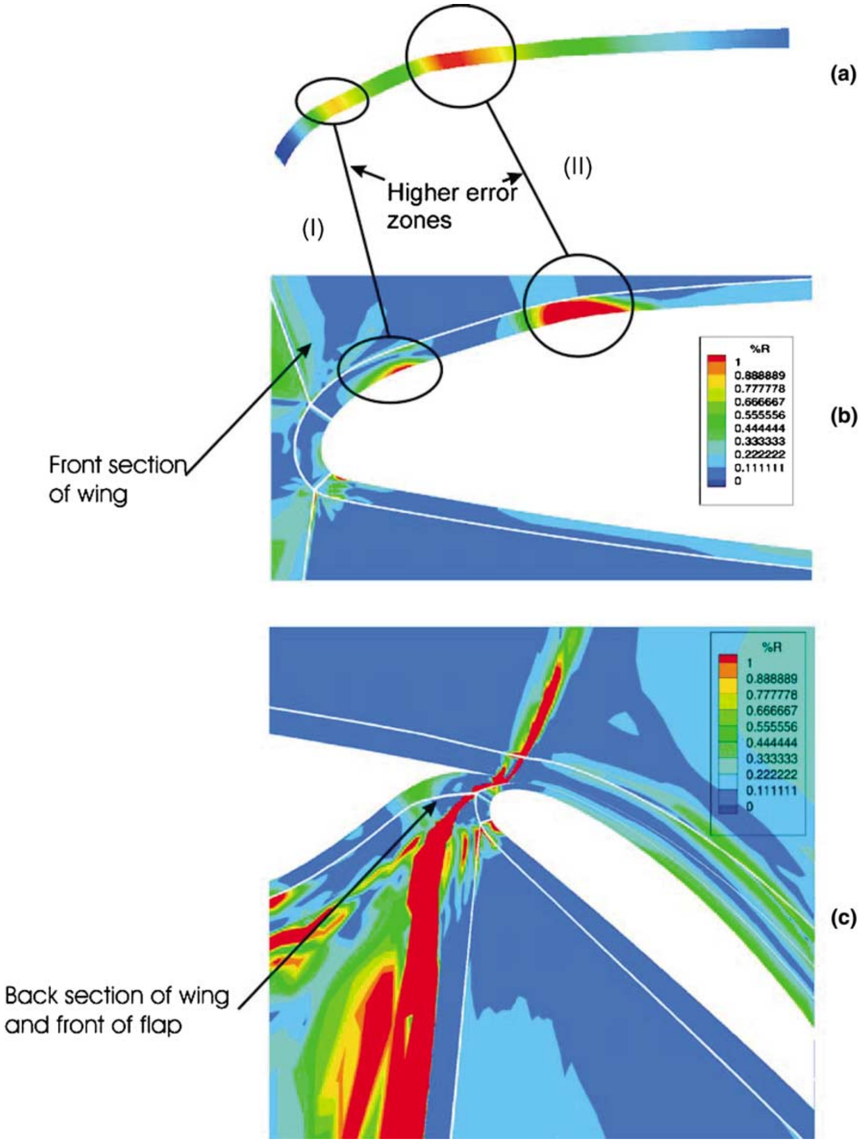

Fig. 15. Eikonal equation residual contours for: (a) Eikonal solution and (b,c) search procedure.

initial and severely deformed aeroelasticity calculation surface meshes. Frames (a,b) are three-dimensional views and (c,d) two-dimensional y–z plane views. For moving mesh performance studies 10 approximately equi-spaced deflection increments between the Fig. 17 extremes are considered. These large increments and the strongly deformed Figs. 17(b) and (d) grids are intended to more severely test the Poisson method. On a 300-mHz SGI Origin, with the present 1.2 million-node mesh, the efficient search procedure takes 57 s. This is around 20% of time step costs. To set up Poisson equation coefficients needs 8.5 s. However, with improved coding, this could be substantially reduced. For the much more challenging steady fixed mesh case the crude solver convergence is adequate requiring 19 s (exactly 33% of that needed for the search). For deflected grid studies, just around 0.2 s solver convergence time is needed to give a convergence error in d of

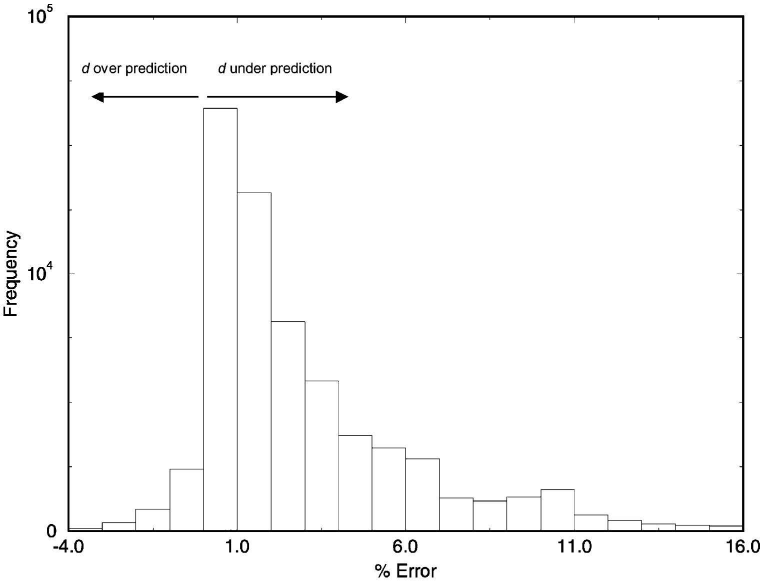

Fig. 16. Double-delta configuration error histogram.

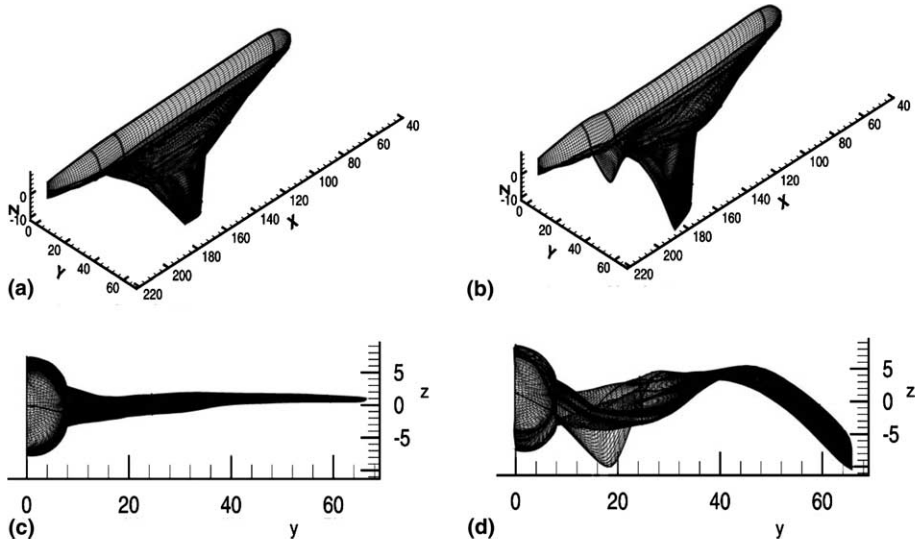

Fig. 17. Initial and fully deformed meshes for aeroelasticity calculation: (a,b) three-dimensional views and (c,d) two-dimensional views.

around 0.1%. This fast crude solver convergence is perhaps partly because near surfaces the grid re-distribution scheme makes temporal changes in adjacent grid node positions relatively small. Also, the Poisson equation solver neglects cross-derivatives. Hence, there is no-solution sensitivity to near surface rotation of grid lines. However, even accounting for these points, results suggest the Poisson method is effective.

The search procedure could, like the Poisson method, probably also be further optimised. One possibility could be not to have a full search for every time step but just to periodically initiate a full search. For intermediate steps perturbed distances could be used. However, the intermediate step duration will be case-dependent and so needs to be chosen with care. For solution adaptive meshes, clearly differential equation-based d methods show great potential. Existing adaptive grid-related code utilities for solving the flow equations could be readily adapted to solve, for example, the simple Poisson equation. To improve Poisson method performance, the Poisson equation could just be solved in surface grid blocks. However, again such a strategy would need to be used with care.

Encouragingly, the Case (d), CFL3D, SA model-based lift, and drag coefficients for the Poisson method are within 0.03% and 0.06% of those for the search procedure. These values being for $Re = 2.2 \times 10^6$ (based on the half wing span), $Ma = 0.96$ and a modest circa 1 million cell grid.

4. Conclusions

The Poisson method distance algorithm is around twice as fast as the search procedure. With parallel and vector processing this could be improved. Although less accurate than the Eikonal, Poisson method-based flow solutions are extremely close to those using a search procedure. For moving grids that do not preserve grid topology the Poisson method is much faster than the search procedure. It is possible to solve the Eikonal equation on highly stretched non-orthogonal curvilinear grids. However, this hyperbolic-natured equation is not straightforward to economically solve and can display instability. For accuracy, Eikonal metrics must be upwinded in the front propagation direction. Addition of a distance scaled Laplacian to the Eikonal equation (giving a Hamilton–Jacobi approach) gives beneficial wall distance properties that can reduce the 'thin wire problem'. The Hamilton–Jacobi equation can be arranged in a form reminiscent of the Navier–Stokes equations. When solved in this form using central differences, as might be expected, solutions diverge. Use of upwinding will hopefully be the subject of future work. However, in any attempt to solve the Hamilton–Jacobi equation in a 'Navier–Stokes' form, the complexity of including metric upwinding will be an issue. For well-optimised fixed grid RANS solutions, the search procedure d evaluation is typically just under 1% of the total solution time. For DES and URANS this fraction would be even less. Therefore, in these cases differential approaches can only really impact through accuracy improving d modifications. For the case (d) aeroelasticity URANS calculation, the search procedure takes around 20% of the solution time. Under these circumstances differential approaches can improve both efficiency and accuracy. Case (d), simulations can take thousands of hours. Liberating a fraction of this leaves significant computational resource for other uses.

Acknowledgements

The present work was carried out at Boeing Commercial Aeroplanes, Seattle amid a group of exceptionally kind people. It was mostly funded through a Royal Academy of Engineering Secondment Award. I am very grateful for this award and to P.R. Spalart for making the visit possible and his very generous encouragement and support during the visit. I would like to thank, L. Hedges, M. Hong, C. Rumsey, J. Sethain, M. Shur, M. Strelets, L. Wigton, and V. Venkatakrishnan for their kind help. Especial thanks are due to C. Hilmes. Other members of Boeing staff have also been very helpful and I apologise if I have forgot to acknowledge you.

References

[1] B.S. Baldwin, H. Lomax, Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows, AIAA Paper 78-257, 1978.

[2] H.C. Chen, V.C. Patel, AIAA J. 26 (6) (1988) 641.

[3] B.S. Baldwin, T.J. Barth, A One-Equation Turbulence Transport Model for High Reynolds Number Wall-Bounded Flows, AIAA Paper 91-0610, 1991.

[4] P.R. Spalart, S.R. Allmaras, La Rech. Aerospatiale 1 (1994) 5.

[5] M. Wolfshtein, Int. J. Heat Mass Transfer 12 (1969) 301.

[6] N. Secundov, M.K.H. Strelets, A.K. Travin, ASME J. Fluids Engrg. 123 (2001) 111.

[7] F.R. Menter, Zonal Two Equation $k - \omega$ Turbulence Models for Aerodynamic Flows, AIAA Paper 93-2906, 1993.

[8] D.B. Spalding, 10th International Heat Transfer Conference, Brighton, UK, 1994 (unpublished).

[9] S.H. Kim, M.K. Chung, AIAA J. 39 (9) (2001) 1803.

[10] P.G. Tucker, ASME J. Fluids Engrg. 123 (June) (2001) 372.

[11] R. Abid, Int. J. Engrg. Sci. 31 (6) (1993) 831.

[12] H. Iacovides, J.W. Chew, Int. J. Heat Fluid Flow 14 (2) (1993) 146.

[13] M. Shur, P.R. Spalart, M. Strelets, A. Travin, in: Proceedings of the 4th International Symposium on Engineering Turbulence Modelling and Measurements, Ajaccio, Corsica, France, vol. 669, 1999.

[14] L.B. Wigton, in: D.A. Caughey, M.M. Hafez (Eds.), Frontiers of Computational Fluid Dynamics, World Scientific, Singapore, 1998, pp. 1–15, Chapter 16; ISBN 981-02-3707-3.

[15] D.A. Boger, AIAA J. 39 (12) (2001) 2404.

[16] P.R. Spalart, Trends in Turbulence Treatments, AIAA Paper 2000-2306, 2000.

[17] E. Fares, W. Schroder, Int. J. Numer. Methods Fluids 39 (2002) 743.

[18] G. Mompean, S. Gavrilakis, L. Machiels, M.O. Deville, Phys. Fluids 8 (7) (1996) 1856.

[19] B.E. Launder, G.J. Reece, W. Rodi, J. Fluid Mech. 68 (Part 3) (1975) 537.

[20] S. Osher, J.A. Sethain, J. Comput. Phys. 79 (1988) 12.

[21] P.G. Tucker, Int. J. Numer. Fluids 33 (2000) 869.

[22] P.G. Tucker, Appl. Math. Model. 22 (1998) 293.

[23] J.A. Sethian, SIAM Rev. 41 (2) (1999) 199.

[24] J.A. Sethian, personal communication, 2002.

[25] M. Strelets, Detached Eddy Simulation of Massively Separated Flows, AIAA Paper 2001-0879, 2001.

[26] L. Rumsey, M.D. Sanetrix, R.T. Biedron, N.D. Melson, E.B. Parlette, Comput. Fluids 25 (2) (1996) 217.

[27] P.G. Tucker, L. Davidson, Zonal *k–l*-Based Large Eddy Simulations, AIAA Paper 2003-0082, 2003.

[28] P.G. Tucker, Novel MILES computations for jet flows with noise, in: 3rd International Symposium on Turbulence and Shear Flow Phenomena, Sendai, Japan, June 25–27, 2003 (accepted).

[29] Y.R. Tsai, J. Comput. Phys. 178 (2002) 175.

[30] P.G. Tucker, Z. Pan, Appl. Math. Model. 24 (2000) 591.

[31] R.E. Bartels, J. Aircraft 37 (3) (2000) 521.